IS 6733: Deep Learning
Homework 3

P1 (40pt): In the code example of "Classifying movie reviews" explained in Lecture 6, make the following changes sequentially to the two neural network models in the example:

1. Change the number of neurons on the two hidden layers to 32 units. (10pt)
2. Use the *tanh* activation (an activation that was popular in the early days of neural networks) instead of *relu* for the two hidden layers. (10pt)
3. Add an additional hidden layer with 32 units and *tanh* activation function. (10pt)

Retrain the newly defined models and evaluate the trained models on the testing dataset to get the accuracy. (10pt)

P2 (60pt): Write a Python code in Colab using NumPy, Panda, Scikit-Learn and Keras to complete the following tasks:

1. Import the Auto MPG dataset using pandas.read_csv(), use the attribute names as explained in the dataset description as the column names, view the strings '?' as the missing value, and whitespace (i.e., '\s+') as the column delimiter. Print out the shape and first 5 rows of the DataFrame. (5pt)
   - Dataset source file: http://archive.ics.uci.edu/ml/machine-learning-databases/auto-mpg/auto-mpg.data
   - Dataset description: http://archive.ics.uci.edu/ml/datasets/Auto+MPG
2. Delete the "car_name" column using .drop() and drop the rows containing NULL value using .dropna(). Print out the shape of the DataFrame. (5pt)
3. For the 'origin' column with categorical attribute, replace it with the columns with numerical attributes using one-hot encoding. Print out the shape and first 5 rows of the new DataFrame. (5pt)
4. Separate the "mpg" column from other columns and view it as the label vector and others as the feature matrix. Split the data into a training set (80%) and testing set (20%) using train_test_split and print out their shapes. Print out the statistics of your training feature matrix using .describe(). (5pt)
5. Normalize the feature columns in both training and testing datasets so that their means equal to zero and variances equal to one. Note that the testing set can only be scaled by the mean and standard deviation values obtained from the training set. Describe the statistics of your normalized feature matrix of training dataset using .describe() in Pandas. (5pt)
   - Option 1: You can follow the normalization steps in the code example of "Predicting house prices: a regression example" in Lecture 6.
   - Option 2: You can use StandardScaler() in Scikit-Learn as in Homework 2 but you may need to transform a NumPy array back to Pandas DataFrame using pd.DataFrame() before calling .describe().
6. Build a sequential neural network model in Keras with two densely connected hidden layers (32 neurons and ReLU activation function for each hidden layer), and an output layer that returns a single, continuous value. Print out the model summary using .summary(). (10pt)
   - Hint: You can follow the "Classifying movie reviews" example in Lecture 6, but need to change input_shape and last layer activation function correctly in the model definition.
7. Define the appropriate loss function, optimizer, and metrics for this specific problem and compile the NN model. (10pt)

8. Put aside 20% of the normalized training data as the validation dataset by setting validation_split = 0.2 and set verbose = 0 to compress the model training status in Keras .fit(). Train the NN model for 100 epochs and batch size of 32 and plot the training and validation loss progress with respect to the epoch number. (10pt)
   - Remember to use GPU for training in Colab. Otherwise, you may find out of memory error or slow execution.
   - There is no need to do K-fold cross-validation for this step.
9. Use the trained NN model to make predictions on the normalized testing dataset and observe the prediction error. (5pt)